

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-011343

(43)Date of publication of application : 16.01.1998

(51)Int.Cl.

G06F 12/00

G06F 17/30

(21)Application number : 08-158337

(71)Applicant : CANON INC

(22)Date of filing :

19.06.1996

(72)Inventor : ENOKIDA MIYUKI

(54) PROCESSOR AND METHOD FOR INFORMATION PROCESSING

(57)Abstract:

PROBLEM TO BE SOLVED: To easily register a file name to which information on a key word, etc., showing the contents of a data file is added when the data file is stored.

SOLUTION: A character string specifying a stored data file among file names given to data files stored in a storage medium is generated and stored in a 0-th column 10. Here, the generated character string is, for example, a number in increasing order, etc. Further, when desired character strings in a key word list are specified as key words, they are stored in a 1st column 11 - a 3rd column 13. Then the character strings stored in the 1st column 11 -

	10	11	12	13	14
登録名	000001	0001	0002	0003	0004
登録日	000002	0002	0003	0004	0005
登録地	000003	0003	0004	0005	0006
登録者	000004	0004	0005	0006	0007

3rd column 13 are added to the character string stored in the 0-th column 10 and they are registered as an actual file name.

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平10-11343

(43)公開日 平成10年(1998) 1月16日

(51)Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 12/00	5 2 0		G 0 6 F 12/00	5 2 0 E
				5 2 0 A
17/30			15/401	3 1 0 A

審査請求 未請求 請求項の数29 O L (全 21 頁)

(21)出願番号 特願平8-158337

(22)出願日 平成8年(1996) 6月19日

(71)出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72)発明者 榎田 幸

東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(74)代理人 弁理士 大塚 康德 (外1名)

(54)【発明の名称】 情報処理装置及び方法

(57)【要約】

【課題】データファイルを蓄積する際に、データファイルの内容を示すキーワード等の情報を付加したファイル名を容易に登録可能とする。

【解決手段】記憶媒体に格納されたデータファイルに付与されるファイル名のうち、格納されたデータファイルを特定する文字列を発生し、これを第0カラム10に登録する。ここで発生する文字列としては、例えば、昇順の番号等が挙げられる。また、キーワードとして、例えばキーワード一覧表の中から所望の文字列を指定すると、それらが第1カラム11～第3カラム13に格納される。そして、第0カラム10に格納された文字列に、第1カラム11～第3カラム13に格納された文字列を付加し、これを実際のファイル名として登録する。

	10	11	12	13	14
第0カラム	第1カラム	第2カラム	第3カラム	登録のファイル名	
画像A	aaa0001	写真	人学式	aaa001	写真_子供_人学式.jpg
画像B	aaa0002	写真	運動会	aaa002	写真_子供_運動会.jpg
画像C	aaa0003	写真	遊園地	aaa003	写真_子供_遊園地.jpg
画像D	aaa0004	カタログ	ハワイ	aaa004	カタログ_旅行_ハワイ.jpg

【特許請求の範囲】

【請求項 1】 データファイルを格納する格納手段と、前記格納手段に格納されたデータファイルを特定する文字列を発生する発生手段と、前記発生手段で発生した文字列に所望の文字列を付加する付加手段と、前記付加手段で得られた文字列を前記データファイルのファイル名として登録する登録手段とを備えることを特徴とする情報処理装置。

【請求項 2】 前記付加手段は、前記データファイルを特定する文字列に付加すべき文字列の候補を表示する表示手段と、前記表示手段によって表示された文字列の候補より所望の候補を指定する指定手段とを備え、前記指定手段で指定された文字列の候補を前記発生手段で発生した文字列に付加することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】 前記付加手段において、前記指定手段は複数の候補を指定し、指定された複数の候補を接続したものを前記発生手段で発生した文字列に付加することを特徴とする請求項 2 に記載の情報処理装置。

【請求項 4】 前記付加手段は、入力された所望の文字列を前記発生手段で発生した文字列に付加することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 5】 前記付加手段は、前記発生手段で発生した文字列と所望の文字列とを所定の記号で接続することを特徴とする請求項 1 乃至 4 のいずれかに記載の情報処理装置。

【請求項 6】 前記付加手段で付加された文字列を用いて、前記格納手段に格納されたデータファイルを検索する検索手段を更に備えることを特徴とする請求項 1 乃至 5 のいずれかに記載の情報処理装置。

【請求項 7】 ディレクトリ構造で画像データファイルを管理する管理手段と、指定されたディレクトリに存在する名称データがファイル名かディレクトリ名かを判断する判断手段と、前記判断手段により名称データがファイル名と判断された場合、当該ファイル名に対応する画像データファイルに基づいて画像を表示する第 1 表示手段と、前記判断手段により名称データがディレクトリ名と判断された場合、当該ディレクトリ名に対応する情報を表示する第 2 表示手段とを備えることを特徴とする情報処理装置。

【請求項 8】 前記第 1 表示手段は、前記ファイル名に対応するデータファイルの符号化画像データを所定サイズの画像へ復元し、表示することを特徴とする請求項 7 に記載の情報処理装置。

【請求項 9】 前記第 1 表示手段は、前記ファイル名に対応するデータファイルの符号化画像データが動画データであった場合、その所定フレームを所定サイズの画

像へ復元し、表示することを特徴とする請求項 7 又は 8 に記載の情報処理装置。

【請求項 10】 前記第 2 表示手段は、前記ディレクトリ名に対応するディレクトリを示すアイコンを表示することを特徴とする請求項 7 に記載の情報処理装置。

【請求項 11】 前記第 2 表示手段は、前記ディレクトリ名に対応するディレクトリ内のデータファイル数をカウントし、前記アイコンとともにそのカウント結果を表示することを特徴とする請求項 10 に記載の情報処理装置。

【請求項 12】 前記第 1 表示手段における前記所定サイズは、当該符号化画像データの原画像よりも小さいことを特徴とする請求項 8 に記載の情報処理装置。

【請求項 13】 前記符号化画像データは、白黒 2 値画像データを J B I G 符号化方式で、白黒多値もしくはカラー画像データを J P E G 符号化方式で符号化して得られる符号化画像データであることを特徴とする請求項 12 に記載の情報処理装置。

【請求項 14】 前記第 1 表示手段は、J P E G 符号化データを所定サイズの画像へ復元する際に、縦横 1 / 8 サイズの画像を復元することを特徴とする請求項 13 に記載の情報処理装置。

【請求項 15】 ディレクトリ構造で画像データファイルを管理する管理手段と、所定の検索条件に適合する画像データファイルについて前記ディレクトリ構造に基づくディレクトリパス名の付加されたファイル名を獲得する獲得手段と、前記獲得手段で獲得したファイル名に付加されたディレクトリパス名に基づいて、当該画像データファイルが現在表示中のディレクトリに存在するか否かを判断する判断手段と、

前記判断手段により前記画像データファイルが現在表示中のディレクトリに存在すると判断された場合、該画像データファイルに基づいて画像を表示する第 1 表示手段と、前記判断手段により前記画像データファイルが現在表示中のディレクトリに存在しないと判定された場合、該画像データファイルが存在するディレクトリを示すディレクトリ情報を表示する第 2 表示手段とを備えることを特徴とする情報処理装置。

【請求項 16】 符号化画像データを復号する復号手段と、指定された符号化画像データを前記復号手段で復号化できるか否かを判断する判断手段と、前記判断手段で復号化できないと判断された場合、所定のアプリケーションを起動し、該所定のアプリケーションに前記指定された符号化画像データを復号化させ、その結果の画像データを取得する取得手段と、前記取得手段で取得した画像データを表示する表示手段とを備えることを特徴とする情報処理装置。

【請求項17】 前記取得手段は、複数のアプリケーションを起動可能であり、前記判断手段が前記指定された符号化画像データを復号化できないと判断した場合、該指定された符号化画像データを該複数のアプリケーションの一つを用いて復号化し、その結果の画像データを取得することを特徴とする請求項16に記載の情報処理装置。

【請求項18】 前記取得手段は、前記複数のアプリケーションを起動しておき、前記指定された符号化画像データの復号化画像データを得るべく、該指定された符号化画像データを該複数のアプリケーションへ順次に提供することを特徴とする請求項17に記載の情報処理装置。

【請求項19】 前記取得手段は、前記指定された符号化画像データの復号化画像データを得るべく、前記複数のアプリケーションを順次起動し、該指定された符号化画像データを起動したアプリケーションへ提供することを特徴とする請求項17に記載の情報処理装置。

【請求項20】 前記表示手段は、前記取得手段で取得した画像データを所定の大きさの画像に変換して表示することを特徴とする請求項16に記載の情報処理装置。

【請求項21】 前記符号化画像データは、静止画像をJBIG方式もしくはJPEG方式で符号化し、動画画像をMPEG方式で符号化することを特徴とする請求項16に記載の情報処理装置。

【請求項22】 データファイルを格納する格納工程と、
前記格納工程に格納されたデータファイルを特定する文字列を発生する発生工程と、
前記発生工程で発生した文字列に所望の文字列を付加する付加工程と、
前記付加工程で得られた文字列を前記データファイルのファイル名として登録する登録工程とを備えることを特徴とする情報処理方法。

【請求項23】 ディレクトリ構造で画像データファイルを管理する管理工程と、
指定されたディレクトリに存在する名称データがファイル名かディレクトリ名かを判断する判断工程と、
前記判断工程により名称データがファイル名と判断された場合、当該ファイル名に対応する画像データファイルに基づいて画像を表示する第1表示工程と、
前記判断工程により名称データがディレクトリ名と判断された場合、当該ディレクトリ名に対応する情報を表示する第2表示工程とを備えることを特徴とする情報処理方法。

【請求項24】 ディレクトリ構造で画像データファイルを管理する管理工程と、
所定の検索条件に適合する画像データファイルについて前記ディレクトリ構造に基づくディレクトリパス名の付加されたファイル名を獲得する獲得工程と、

前記獲得工程で獲得したファイル名に付加されたディレクトリパス名に基づいて、当該画像データファイルが現在表示中のディレクトリに存在するか否かを判断する判断工程と、

前記判断工程により前記画像データファイルが現在表示中のディレクトリに存在すると判断された場合、該画像データファイルに基づいて画像を表示する第1表示工程と、

前記判断工程により前記画像データファイルが現在表示中のディレクトリに存在しないと判定された場合、該画像データファイルが存在するディレクトリを示すディレクトリ情報を表示する第2表示工程とを備えることを特徴とする情報処理方法。

【請求項25】 符号化画像データを復号する復号工程と、

指定された符号化画像データを前記復号工程で復号化できるか否かを判断する判断工程と、

前記判断工程で復号化できないと判断された場合、所定のアプリケーションを起動し、該所定のアプリケーションに前記指定された符号化画像データを復号化させ、その結果の画像データを取得する取得工程と、

前記取得工程で取得した画像データを表示する表示工程とを備えることを特徴とする情報処理方法。

【請求項26】 データファイルの記憶媒体への格納を管理するための制御プログラムを格納するコンピュータ可読メモリであって、

前記記憶媒体に格納されたデータファイルを特定する文字列を発生する発生工程のコードと、

前記発生工程で発生した文字列に所望の文字列を付加する付加工程のコードと、

前記付加工程で得られた文字列を前記データファイルのファイル名として登録する登録工程のコードとを備えることを特徴とするコンピュータ可読メモリ。

【請求項27】 ディレクトリ構造で記憶媒体に格納された画像データファイルを管理する制御プログラムを格納するコンピュータ可読メモリであって、

指定されたディレクトリに存在する名称データがファイル名かディレクトリ名かを判断する判断工程のコードと、

前記判断工程により名称データがファイル名と判断された場合、当該ファイル名に対応する画像データファイルに基づいて画像を表示する第1表示工程のコードと、
前記判断工程により名称データがディレクトリ名と判断された場合、当該ディレクトリ名に対応する情報を表示する第2表示工程のコードとを備えることを特徴とするコンピュータ可読メモリ。

【請求項28】 ディレクトリ構造で記憶媒体に格納された画像データファイルを管理する制御プログラムを格納するコンピュータ可読メモリであって、

所定の検索条件に適合する画像データファイルについて

前記ディレクトリ構造に基づくディレクトリパス名の付加されたファイル名を獲得する獲得工程のコードと、前記獲得工程で獲得したファイル名に付加されたディレクトリパス名に基づいて、当該画像データファイルが現在表示中のディレクトリに存在するか否かを判断する判断工程のコードと、

前記判断工程により前記画像データファイルが現在表示中のディレクトリに存在すると判断された場合、該画像データファイルに基づいて画像を表示する第 1 表示工程のコードと、

前記判断工程により前記画像データファイルが現在表示中のディレクトリに存在しないと判定された場合、該画像データファイルが存在するディレクトリを示すディレクトリ情報を表示する第 2 表示工程のコードとを備えることを特徴とするコンピュータ可読メモリ。

【請求項 29】 符号化画像データを復号化して表示するための制御プログラムを格納するコンピュータ可読メモリであって、

符号化画像データを復号する復号工程のコードと、指定された符号化画像データを前記復号工程で復号化できるか否かを判断する判断工程のコードと、

前記判断工程で復号化できないと判断された場合、所定のアプリケーションを起動し、該所定のアプリケーションに前記指定された符号化画像データを復号化させ、その結果の画像データを取得する取得工程のコードと、前記取得工程で取得した画像データを表示する表示工程のコードとを備えることを特徴とするコンピュータ可読メモリ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、符号化された複数のデータファイルを管理する情報処理装置及びその方法に関する。

【0002】

【従来の技術】従来より、例えば画像データ等をデータファイルとして、夫々を特定するファイル名を付与して蓄積する装置がある。この種の装置では、そのファイル名を用いて所望のデータファイルを特定したり、検索したりすることが可能である。

【0003】この種のデータファイル管理を実現するものとして、DOS（商標）／Windows 3.1（商標）などのファイル管理システムが知られている。これらのファイル管理システムでは、8文字のファイル名と3文字の属性で1つのファイル名を構成する。

【0004】また、従来、複数ファイルからキーワード検索などの検索方法等で検索された複数のファイルを表示する場合は、全てのファイルを並列に1度に表示するように構成されている。ここで、画面の表示スペースや検索されたファイルの数などにより、1度に検索された全てのファイルを表示できない場合には、画面のスクロ

ールやページ切り替え等の方法により順次表示することが可能となっている。また画像符号化データが、静止画像符号化データと、動画符号化データの2種類のデータがある場合には、各々検索表示画像を表示する方法が異なるため、各々別々に表示するように構成されている。

【0005】また、従来、J P E G等の符号化方式で符号化されている画像ファイルをデコード／表示する場合は、アプリケーションがサポートしている符号化モードで符号化された符号化データ・ファイルのみをデコード／表示するように構成されている。

【0006】また、ある符号化方式そのものをサポートしていない場合に、そのサポートしていない符号化データ・ファイルをデコード／表示する場合は、当該アプリケーションとは別のその符号化方式のデコード／表示機能を有する他のアプリケーションを起動し、当該他のアプリケーションでデコード／表示するように構成されている。

【0007】

【発明が解決しようとする課題】しかしながら、上記のデータファイル管理システムでは、8文字までのファイル名ではその画像の特徴などを表現することができず、画像の特徴などを記述する場合には別のファイルを用意しなければならないという問題がある。例えば、1つの画像データに対して実際の画像データを含むデータファイルと、画像の特徴などを記述する検索用のキーワード情報ファイルとを別々に管理する必要が生じることになる。この場合、例えば、画像データファイルのみを移動や消去したりした場合には、対応する画像データファイルの存在しないキーワード情報ファイルのみが残されることになる。或は、逆に、対応するキーワード情報ファイルを持たない画像データファイルが存在してしまうという問題が生じる。

【0008】更に、画像データファイルの検索時にキーワード検索などを行う場合には、各画像データに対して、検索情報を格納したファイルをオープンして、キーワード検索並びにフル・テキスト・サーチ等の処理を行う必要があり、処理が複雑で重くなるという問題がある。

【0009】また、各ファイル個別にこのようなキーワード情報ファイルを管理しない場合には、管理／検索専用プログラムを用意し、このプログラムの基で、キーワード情報を管理しなければならないという問題がある。

【0010】また、上記従来例による検索画像データの表示方法によれば、ディレクトリ構造でファイルの種類などを管理している場合にも、検索された全ての画像データファイルが並列に表示されることになる。即ち、ディレクトリ構造が全く反映されず、検索されたファイルの絞り込みがしにくくなるという問題がある。この場合、ディレクトリ構造で画像データファイルを管理していることが全く無意味になってしまう。

【0011】特に、通常はファイル名のみを表示するため、別々のディレクトリに同じ名前のファイルが存在するときには、それらのファイルを特定することが困難となる。更に格納されているファイルの数が多くなると、ファイル名からでは、ユーザが実際に検索したいファイルを特定することが困難となってしまう。また符号化された画像データファイル（以下、画像符号化データファイルという）の検索結果として、静止画像符号化データファイルと動画画像符号化データファイルとが混在するような場合には、別々の表示形態による表示が要求されるため、動画像と静止画像を統一的に扱えないという問題がある。

【0012】JPEG符号化アルゴリズムには、図19に示すように1つの符号化アルゴリズムの中でも複数の方式があり、必須機能は、253で示す「ベースライン・プロセス」のみで、他の方式、機能は、オプションである。すなわちデコードの機能として、オプションの部分は、サポートしてもしなくてもどちらでも良い部分である。

【0013】そのため、上記従来例では、例えばJPEG符号化方式のデコード／表示をサポートしている場合、同じJPEG方式のデータ・ファイルでも、デコード／表示できるファイルとできないファイルがあるという問題がある。

【0014】また本アプリケーションがサポートしていない符号化方式で符号化されている画像データ・ファイルをデコード／表示する際に別アプリケーションを起動する場合は、その別アプリケーションがデコード／表示を行うため、ユーザインターフェースを含め、本アプリケーションとは全く異なった動作を行うことになる。このため、特に本アプリケーションが独自のユーザ・インターフェース等の機能を持っている場合には、本アプリケーションの機能が全く使用できなくなるという問題がある。

【0015】本発明は、上記の問題に鑑みてなされたものであり、データファイルを蓄積する際に、データファイルの内容を示すキーワード等の情報を付加したファイル名を容易に登録できる情報処理装置及びその方法を提供することを第1の目的とする。

【0016】また、本発明の第2の目的は、ファイル名として付与されたキーワードを、データファイルの検索情報として用いることを可能とする情報処理装置及びその方法を提供することにある。

【0017】また、本発明の第3の目的は、複数の画像データファイルについて、各画像データファイル内の画像データに基づく表示を行うとともに、各画像データファイルが存在するディレクトリ名等のディレクトリ情報を表示することを可能とし、画像データファイルのディレクトリ構造を忠実に表示すると共に、各画像データを直接表示することを可能とすることにある。

【0018】更に本発明の第4の目的は、アプリケーションがサポートしている画像符号化方式でデコードできない画像符号化データ・ファイルをデコード／表示する際に、その画像符号化データ・ファイルを別アプリケーションでデコードさせ、その結果を当該アプリケーションによって表示することを可能とし、当該アプリケーションが備えるユーザインターフェース機能等を有効とする情報処理装置及びその方法を提供することにある。

【0019】

10 【課題を解決するための手段】上記の第1の目的を達成するための本発明の情報処理装置は以下の構成を備える。即ち、データファイルを格納する格納手段と、前記格納手段に格納されたデータファイルを特定する文字列を発生する発生手段と、前記発生手段で発生した文字列に所望の文字列を付加する付加手段と、前記付加手段で得られた文字列を前記データファイルのファイル名として登録する登録手段とを備える。

20 【0020】また、好ましくは、前記付加手段は、前記データファイルを特定する文字列に付加すべき文字列の候補を表示する表示手段と、前記表示手段によって表示された文字列の候補より所望の候補を指定する指定手段とを備え、前記指定手段で指定された文字列の候補を前記発生手段で発生した文字列に付加する。付加すべき文字列の候補を表示してユーザが選択するので、容易にしかも適切に文字列を付加できるからである。特に、例えば、ファイル名で検索を行うような場合に、検索用のキーワードとして一貫した文字列を付加できるようになる。

30 【0021】また、好ましくは、前記付加手段における前記指定手段は複数の候補を指定し、指定された複数の候補を接続したものを前記発生手段で発生した文字列に付加する。複数の文字列を接続することが可能となり、より多様なファイル名の付与を行えるからである。

【0022】また、好ましくは、前記付加手段は、入力された所望の文字列を前記発生手段で発生した文字列に付加する。所望の文字列を自由に付加できるようになるからである。

【0023】また、好ましくは、前記付加手段は、前記発生手段で発生した文字列と所望の文字列とを所定の記号で接続する。付加された文字列とデータファイルを特定する文字列を容易に分離できる様になるからである。このため、例えば、付加された文字列を用いてキーワード検索を行うような場合に、処理が容易になる。

【0024】また、好ましくは、前記付加手段で付加された文字列を用いて、前記格納手段に格納されたデータファイルを検索する検索手段を更に備える。上記第2の目的が達成される。

50 【0025】また、上記第3の目的を達成するための本発明の情報処理装置は、ディレクトリ構造で画像データファイルを管理する管理手段と、指定されたディレク

トリに存在する名称データがファイル名かディレクトリ名かを判断する判断手段と、前記判断手段により名称データがファイル名と判断された場合、当該ファイル名に対応する画像データファイルに基づいて画像を表示する第1表示手段と、前記判断手段により名称データがディレクトリ名と判断された場合、当該ディレクトリ名に対応する情報を表示する第2表示手段とを備える。

【0026】また、好ましくは、前記第1表示手段は、前記ファイル名に対応するデータファイルの符号化画像データを所定サイズの画像へ復元し、表示する。

【0027】また、好ましくは、前記第1表示手段は、前記ファイル名に対応するデータファイルの符号化画像データが動画像データであった場合、その所定フレームを所定サイズの画像へ復元し、表示する。例えば、検索結果表示において、動画像データと静止画像データが混在するような場合でも、検索された画像データファイルを示すためのイメージ表示を行えるからである。

【0028】また、好ましくは、前記第2表示手段は、前記ディレクトリ名に対応するディレクトリを示すアイコンを表示する。

【0029】また、好ましくは、前記第2表示手段は、前記ディレクトリ名に対応するディレクトリ内のデータファイル数をカウントし、前記アイコンとともにそのカウント結果を表示する。アイコンで表示されたディレクトリ内に存在するデータファイル数が把握できるからである。

【0030】また、好ましくは、前記第1表示手段における前記所定サイズは、当該符号化画像データの原画像よりも小さい。

【0031】また、好ましくは、前記符号化画像データは、白黒2値画像データをJBIG符号化方式で、白黒多値もしくはカラー画像データをJPEG符号化方式で符号化して得られる符号化画像データである。

【0032】また、好ましくは、前記第1表示手段は、JPEG符号化データを所定サイズの画像へ復元する際に、縦横1/8サイズの画像を復元する。

【0033】また、上記第3の目的を達成するための他の構成の情報処理装置は、ディレクトリ構造で画像データファイルを管理する管理手段と、所定の検索条件に適合する画像データファイルについて前記ディレクトリ構造に基づくディレクトリパス名の付加されたファイル名を獲得する獲得手段と、前記獲得手段で獲得したファイル名に付加されたディレクトリパス名に基づいて、当該画像データファイルが現在表示中のディレクトリに存在するか否かを判断する判断手段と、前記判断手段により前記画像データファイルが現在表示中のディレクトリに存在すると判断された場合、該画像データファイルに基づいて画像を表示する第1表示手段と、前記判断手段により前記画像データファイルが現在表示中のディレクトリに存在しないと判定された場合、該画像データファイ

ルが存在するディレクトリを示すディレクトリ情報を表示する第2表示手段とを備える。

【0034】また、上記第4の目的を達成するための本発明の情報処理装置は以下の構成を備える。即ち、符号化画像データを復号する復号手段と、指定された符号化画像データを前記復号手段で復号化できるか否かを判断する判断手段と、前記判断手段で復号化できないと判断された場合、所定のアプリケーションを起動し、該所定のアプリケーションに前記指定された符号化画像データを復号化させ、その結果の画像データを取得する取得手段と、前記取得手段で取得した画像データを表示する表示手段とを備える。

【0035】また、好ましくは、前記取得手段は、複数のアプリケーションを起動可能であり、前記判断手段が前記指定された符号化画像データを復号化できないと判断した場合、該指定された符号化画像データを該複数のアプリケーションの一つを用いて復号化し、その結果の画像データを取得する。複数のアプリケーションを用いることにより、復号化が可能な符号化画像データの範囲が広がるからである。

【0036】また、好ましくは、前記取得手段は、前記複数のアプリケーションを起動しておき、前記指定された符号化画像データの復号化画像データを得るべく、該指定された符号化画像データを該複数のアプリケーションへ順次に提供する。

【0037】また、好ましくは、前記取得手段は、前記指定された符号化画像データの復号化画像データを得るべく、前記複数のアプリケーションを順次起動し、該指定された符号化画像データを起動したアプリケーションへ提供する。

【0038】また、好ましくは、前記表示手段は、前記取得手段で取得した画像データを所定の大きさの画像に変換して表示する。

【0039】また、好ましくは、前記符号化画像データは、静止画像をJBIG方式もしくはJPEG方式で符号化し、動画像をMPEG方式で符号化する。

【0040】

【発明の実施の形態】以下、添付の図面を参照し本発明の好適な実施形態を説明する。

【0041】〔第1の実施形態〕UNIX（商標）やWindows 95（商標）等のファイル管理システムでは、256バイトまでの長さのファイル名でファイル管理が行える。本実施形態では、このようなファイル管理システムに適用可能なデータ管理装置を説明する。

【0042】図1は第1の実施形態による画像データ管理装置の構成を示すブロック図である。図において、1はCPUでありROM2或はRAM3に格納された制御プログラムを実行することにより、各種の制御を実現する。2はROMであり、当該画像データ管理装置の電源投入時においてCPU1が実行するブートプログラム

や、各種データを格納する。3はRAMであり、後述のフローチャートで説明する各種制御プログラムをロードして格納する。また、RAM3はCPU1が各種の制御を実行するに際しての作業領域を提供する。4は外部記憶部であり、複数の画像データファイルや、制御プログラムを格納する。

【0043】5は入力部であり、当該画像データ管理装置に対する各種指示の入力を行う。入力部5は、例えばキーボードやポインティングデバイス等で構成される。6はディスプレイであり、CPU1の制御によって各種の表示を行う。7はスキャナであり、原稿画像を光学的に読み取って、得られた画像データを画像データ管理装置に提供する。8はバスであり、上述の各構成を相互に接続する。

【0044】図2は、第1の実施形態における画像データファイルのファイル名のフォーマットを示す図である。本第1の実施形態では、ファイル名に付加する検索用のキーワードをキーワード群から選択し、そのキーワードを接続して1つのファイル名を作成する場合を説明する。なお、説明を簡単にするために、本第1の実施形態では、1つのファイル名に付加するキーワードは3つとする。

【0045】このキーワードは、図中の11、12、13で示す第1～第3カラムのフィールドに格納される。第0カラム10は、システム内部で生成する文字列であり、各画像データファイルにユニークな文字列である。このユニークな文字列によって、唯一の画像データファイルを特定することが可能である。また、14は、第0カラム10～第3カラム13に格納された文字列によって形成された、実際のファイル名を示している。

【0046】なお、これら実際のファイル名14は、外部記憶部4のディレクトリ領域に格納され、その実体となる画像データファイルと関連付けられている。このようなディレクトリ領域については周知であるので、ここでは詳細な説明を省略する。

【0047】以上のような構成を備えた第1の実施形態のデータ管理装置の動作について図3～図5を参照して説明する。

【0048】図3は画像データを登録する際の操作画面例を示す図である。20は、画像データ登録用ウィンドウを示す。即ち、外部記憶部4へ画像データをデータファイルとして格納する際のファイル名の設定に際して、画像データ登録用ウィンドウ20がディスプレイ6に表示される。21はキーワード選択領域であり、ファイル名として付加するキーワード群を表示する。22は登録ボタンであり、データファイルの登録作業の開始等を指示するためのものである。ユーザは、入力部5を操作して、図3に示したような画像データ登録用ウィンドウ20のキーワード選択領域21より所望のキーワードを選択することになる。

【0049】次に、ユーザがスキャナ7等の画像入力装置から画像を入力し、その画像データを外部記憶部4にファイル名を付与して蓄積する処理を説明する。図4は第1の実施形態による画像データの登録手順を示すフローチャートである。ここで、ユーザは、画像データを格納すべきディレクトリを予め指定しているものとする。次に、ユーザが画像データを画像入力装置から入力すると、図3に示す画像データ登録画面が表示される。図4のフローチャートはこのような状態となったからの制御手順を示している。

【0050】ステップS10において、第0カラム10への文字列の登録を行う。図5はファイル名の第0カラムへ登録すべき文字列を生成する手順を示すフローチャートである。図5のステップS20において、当該画像データファイルを格納するディレクトリ内のデータファイル名を検索し、既に使用されている文字列を検出する。次にステップS21において、未使用の文字列を発生する。

【0051】第0カラムの文字列としては、例えば、図2に示したように、ファイル名にシーケンシャルな番号を付与するようにすれば、より容易にユニークな文字列を発生することが可能となる。図2に示した画像Aの場合は、当該ディレクトリに格納する最初の画像データファイルなので、例えば、“aaa0001”とし、以後、画像B、画像Cと蓄積する場合には、画像Bには“aaa0002”、画像Cには“aaa0003”というように、順次数字を増加させればよい。このようにして、当該ディレクトリ内においてユニークな文字列を容易に生成することができる。

【0052】図4に戻り、第0カラム部分の文字列を登録すると、処理はステップS11へ進む。ステップS11では、この処理部での内部変数「i」を0に初期化する。次にステップS12へ進み、キーワード選択領域21に示されるキーワード群からキーワードを選択する処理に移る。ここでは、キーワード選択領域21に表示されているキーワードの一つを、ユーザがマウス等のポインティングデバイスで選択する。1つのキーワードが選択されると、そのキーワードを記憶し、ステップS13へ進む。

【0053】ステップS13では、内部変数「i」を1つインクリメントし、「i」が3に等しいか否かを判定する。これは、第1の実施形態におけるキーワード数を3個までとしているからであり、「i」が3未満であれば処理をステップS11へ戻し、再度キーワードを選択する。一方、「i」が3に等しければステップS14へ進む。以上のようなステップS12、S13によるキーワード選択処理により、1つの画像データファイルに対して3つのキーワードが選択される。例えば、図2の画像Aでは、「写真」「子供」及び「入学式」とう3つのキーワードが選択されたことを示す。

【0054】ステップS14では、確認のための操作入力を待ち、操作入力検出されるとステップS15へ進む。ステップS15では当該操作入力が登録ボタン22の選択か否かを判断し、登録ボタン22の選択であれば、ステップS16へ進む。ステップS16では、ステップS10で生成した文字列、ステップS12で選択したキーワードを用いて実際のファイル名を生成する。図2の画像Aの場合は、第0カラムが「aaa0001」、第1カラムが「写真」、第2カラムが「子供」、第3カラムが「入学式」となり、これらの各フィールドの文字列を「_」で接続し、「aaa0001_写真_子供_入学式.jpg」というファイル名を生成している。

【0055】なお、このとき、入力画像がカラー原稿の場合は、例えばJPEG方式で画像データを符号化し、属姓名として「.jpg」を付けている。

【0056】最後にステップS17において、入力された画像データをステップS16で作成したファイル名で、且つJPEG方式で符号化する場合には、当該符号化処理を行い、外部記憶部4に蓄積して、画像データ登録処理を終了する。なお、以上の処理を画像B～Dに対して行うことで、例えば、画像Bについては「aaa0002_写真_子供_運動会.jpg」というファイル名が、画像Cについては「aaa0003_写真_子供_遠足.jpg」というファイル名が、画像Dについては「aaa0004_カタログ_旅行_ハワイ.jpg」というファイル名を夫々付与することができる。

【0057】また、以上のようなファイル名で蓄積された画像データファイルの検索では、ファイル名の中に検索時に必要なキーワードが挿入されているため、ファイル名の文字列操作だけでキーワード検索を行うことが可能となる。また、キーワードを「_」によって接続しているため、ファイル名からキーワードを取出す処理も簡単に行える。これにより、容易且つ高速なキーワード検索が実現される。

【0058】〔第2の実施形態〕上記第1の実施形態では、検索情報としてのキーワードを予め決められたキーワード群から選択することを説明したが、キーワードの付与方法はこれに限られるものではない。第2の実施形態では、キーワードを決めない、即ちフリー・キーワードの場合を説明する。

【0059】図6は第2の実施形態における画像データファイルのファイル名のフォーマットを示す図である。第2の実施形態におけるファイル名のフォーマットは、第1の実施形態で示した図2のフィールドのうち、第0カラム10はそのままである。また、第1カラム11'がフリー・キーワード領域として使用され、第2、第3カラムは省かれた形態となる。また、ファイル名の長さは256バイトまでとする。従って、第0カラムに7バイトの文字列と、第0及び第1カラムを接続するのに用いる1バイトの文字「_」分を除く248バイト（25

6-8=248)のフリーキーワードをファイル名の一部として登録できることになる。

【0060】更に、第2の実施形態における画像データ登録画面は、図3に示した登録用ウインドウ20のキーワード選択領域21が、フリーキーワード入力用の文字入力領域となる。

【0061】図7は、第2の実施形態における画像データ登録処理の手順を表すフローチャートである。同図において、第1の実施形態(図4)と同じ処理内容のステップには、同一のステップ番号を付し、それらの処理については詳細な説明を省略する。なお、第1の実施形態と同様に、ユーザは、先ず、画像データを格納するディレクトリ決定しており、登録対象となる画像データをスキャナ7より入力してあるものとする。

【0062】先ず、ステップS10において、当該画像データに対してユニークな文字列を発生し、第0カラム10のフィールドの文字列として決定する。続いて、ステップS24において、本処理の内部変数「i」を8に設定する。次にステップS25へ移り、ユーザが画像データに付加する、画像の特徴などを表すキーワードをキーボード等の入力部5から入力する。

【0063】そして、登録ボタン22が押されていない場合は、ステップS27へ進み、入力された文字数が、ファイル名の最大バイト数である256バイトを越えているか否かを判断する。ここで、ステップS25においてファイル名の一部として入力された文字が、1バイト文字か2バイト文字かに応じて内部変数「i」を増加させることは言うまでもない。

【0064】登録ボタン22が押下されておらず、内部変数「i」が256(バイト)を越えていなければ上記ステップS25～S27の処理が繰り返される。ここで、登録ボタン22が操作されれば、ステップS26からステップS16へ進む。従って、ファイル名の長さは可変となる。また、ステップS25で文字が入力されたことによって、内部変数「i」が256バイトを越えた場合は、ステップS27からステップS28へ進む。ステップS28では、確認のために登録ボタン22が操作されたか否かを判定し、操作されていない場合は登録中止として本処理を終了する。一方、ステップS28において登録ボタン22が操作されればステップS16へ進む。

【0065】以上のようにして、ユーザによって入力されたフリーキーワードは、ステップS16、S17において、ファイル名の一部として用いられ、第1の実施形態と同様な画像データファイルの登録が行われる。

【0066】また、画像データファイルの検索の場合には、第1の実施形態と同様に、ファイル名をフルテキストサーチすれば良く、画像データの検索を高速に行うことが可能となる。

【0067】なお、上記実施形態では、ファイル名の最

大長を256バイトとしたが、これはファイル管理システムに依存するものであり、256バイトに限られるものではない。また、絡む間を接続してファイル名を作成する際に、「_」を挿入して接続したがこれに限られず、別の方式で各カラムのキーワードを接続してもよい。

【0068】また、キーワードとして説明した単語はこれに限るものではなく、更にキーワードの個数を可変としてもよい。また、キーワード選択領域21に表示するキーワードの数、内容は、ユーザによって任意に変更可能である。即ち、ユーザがキーワードをカスタマイズできる構成としてもよいことは明かである。

【0069】また、第0カラムに格納するユニークな文字列の生成方法も、上記実施形態で説明した方式に限られるものではない。

【0070】以上説明したように、上記実施形態によれば、長いファイル名を許すファイル管理システムを有する画像データ管理装置において、検索時に使用するキーワード等の検索情報を挿入してファイル名を作成することが可能となる。このため、画像データを検索する際にはこのファイル名によって検索処理を遂行することが可能となり、キーワード情報ファイル等の余計なファイルをオープンして検索を行うというような処理が不要となる。このため、検索処理を高速に行うことができる。

【0071】また、ファイル名に画像の特徴などの検索情報が挿入されているので、DOS/Windowsにおける「dir」コマンドや、UNIXにおける「ls」等のファイル名を表示するコマンドによって、ある程度の画像データ検索を行うことも可能となる。

【0072】〔第3の実施形態〕上記実施形態では、画像データファイルの検索を高速にしかも容易な構成で実現するデータ管理装置を説明した。以下の、第3～第5の実施形態では、検索された画像データファイル等の表示方法に焦点を当てて説明する。なお、本実施形態における画像データ管理装置の構成は、第1の実施形態（図1）と同様であり、ここでは説明を省略する。

【0073】図8は第3の実施形態における検索画像表示例を示す図である。図中101は検索された画像データファイルを表示するウィンドウである。また、2-1から2-6は、画像データファイルに格納された画像符号化データに基づく画像表示枠を示す。また、3-1と3-2は、ディレクトリ情報を表示するためのディレクトリ表示枠を示す。

【0074】図9は第3の実施形態におけるディレクトリ構造例を示す図である。図9に示したディレクトリ構造によれば、ディレクトリ“DIR-A”の下に“FILE1”から“FILE6”の画像データファイルと、“DIR-1”及び“DIR-2”の2つのディレクトリが存在する。更に、“DIR-1”には“FILE7”及び“FILE8”が、“DIR-2”には“FILE

LE9”及び“FILE10”の2つの画像ファイル夫々存在する。第3の実施形態では、図9の“DIR-A”の下にあるファイルを表示する場合について説明することにする。

【0075】以下、図10～図12のフローチャートに従って本実施形態の動作を説明する。図10は第3の実施形態における検索結果表示処理の手順を示すフローチャートである。また、図11は、第3の実施形態におけるファイル表示処理の手順を表すフローチャートである。さらに、図12は第3の実施形態におけるディレクトリ表示処理の手順を表す図である。

【0076】まず、表示方法の詳細を図10及び図11に示すフローチャートに従って説明する。なお、本実施形態では、画像符号化データとして、白黒2値画像の場合はJBIG方式で階層化された符号化データが、カラー画像或いは白黒多値画像の場合はJPEG方式で符号化された符号化データが格納されているものとする。更に、動画画像の場合は、WindowsのAVIフォーマットで格納された動画画像符号化データが格納される。

【0077】ユーザからディレクトリ“DIR-A”の下に格納されているファイルを表示する旨が指定されたとすると、図11に示すフローチャートが実行される。先ずステップS100にて、指定されたディレクトリ（DIR-A）内に、表示すべき画像符号化データファイル或いはディレクトリが存在するかをチェックする。表示する画像符号化データファイル/ディレクトリが無ければ、ステップS100より本処理を終了する。

【0078】ファイル或はディレクトリが存在する場合は、ステップS100よりステップS101に移る。ステップS101では、表示すべく獲得したファイル名文字列が通常のファイルを示すかどうかを判断する。なお、本実施形態において、ファイル名文字列とは、あるディレクトリに存在するファイル名及びサブディレクトリ名を表す文字列であるとする。

【0079】ステップS101において、当該ファイル名文字列が通常のファイルに対応していればステップS102に進み、後述のファイル表示処理を行う。一方、当該ファイル名文字列が通常のファイルに対応していない場合には、ステップS103に移り、当該ファイル名文字列がディレクトリを示すものかどうかを判断する。ディレクトリを示している場合には、ステップS104に進み、後述のディレクトリ表示処理を行う。以上の処理を指定されたディレクトリ中の全てのファイル名文字列について行う。

【0080】上記の処理では、ファイル名文字列の種類が、通常のファイルかディレクトリかを分類するのみであるので、ファイル名文字列に付加された属性を調べることで判断できる。図9のようなデータファイル構成の場合を例に上げて説明する。図9に示すディレクトリ構造の場合では、先ず“FILE1”は、通常のファイル

なので、図10のステップS101でファイルと判断されステップS102に移る。続く“FILE2”、“FILE3”も同様である。次に、ファイル名文字列として“DIR-1”が獲得されると、これは通常のファイル名ではないので、処理はステップS101からステップS103に進む。”DIR-1”はディレクトリ名であるため、処理はステップS103からステップS104に進む。以上のような処理を最後の“DIR-2”まで行うことになる。

【0081】次にステップS102におけるファイル表示処理の詳細な処理手順を図11を用いて説明する。図11の処理では、画像符号化ファイルがJBIG符号化データか、JPEG符号化データか、Windows-AVI符号化データかを判断して、各デコード処理を行い、表示用の画像データを作成する。

【0082】この判断は、例えば、JPEG符号化方式で符号化された画像符号化ファイルには、“.JPG”を、JBIG符号化方式で符号化された画像符号化ファイルには、“.JBG”を、Windows-AVI符号化方式の画像符号化ファイルには、“.AVI”をフ

ァイルの属性として付与しておけば容易に判断することができる。

【0083】ここで、当該データファイルがJBIG符号化データの場合は、ステップS110の判断でステップS112に移り、JBIGの最低解像度の符号化データを読み出し、デコード処理を行う。JPEG符号化データの場合は、ステップS111の判断でステップS113に移り、JPEGの階層デコード処理を用いてX、Yサイズとも1/8のサイズの画像データのデコード処理を行う(図13により後述する)。更に、当該データ

ファイルがWindows-AVI符号化データの場合には、ステップS116よりステップS117に移り、Windows-AVI符号化データから先頭フレームのみのデコードを行い静止画像を作る。データファイルが上記のいずれにも属さない場合には、そのまま本処理を終了する。

【0084】次にステップS114で、各ステップS112、S113、S117でデコードした画像データを検索画像表示用のサイズになるように画像サイズの調整する処理を行う。即ち、デコードして得られた画像データのサイズが図8で表示する検索画像表示サイズより大きい場合には、表示サイズに納まるように縮小する。一方デコードして得られた画像データの画像サイズが上記表示サイズより小さい場合には、拡大処理を行う。

【0085】以上の処理により、原画像のサイズが異なっている場合でも、図8で示したように同じ大きさの検索画像データとして表示することが出来る。検索表示画像をウィンドウに表示した後、ステップS115にて、その画像ファイルのファイル名(ファイル名文字列)を表示する。

【0086】次に、上記ステップS104におけるディレクトリ表示処理について図12のフローチャートを参照して説明する。本実施形態の場合には、ディレクトリ用のアイコンを表示することを考える。この場合ステップS120において、ディレクトリ用のアイコンを表示し、ステップS121にて、ディレクトリ名(当該ファイル名文字列)を表示する。

【0087】その後、例えばユーザが、“DIR-1”の下にあるファイルを表示したいことを指示した場合には、上記の処理をディレクトリ“DIR-1”に対して行えば良い。“DIR-2”についても同様である。

【0088】図13は、JPEG1/8サイズのデコード処理の処理手順を説明する図である。図13において、90は通常のJPEG符号化データである。この符号化データ90に対して、ハフマン・デコード部91が、通常のハフマン・デコードを施す。そして、DCT係数部92により、デコードされた画像データの8×8のDCT係数を得る。このDCT係数を1×1RDCT部93により、逆DCT変換し、X、Yサイズとも1/8の画像に復元する。

【0089】以上説明したように、第3の実施形態によれば、指定されたディレクトリの画像データファイルを所定の大きさの枠内に表示するとともに、当該ディレクトリにおけるディレクトリがアイコン及びディレクトリ名で表示される。

【0090】なお、上記実施形態において、画像データファイルとそれ以外のデータファイル(例えば文書データファイル)が同じディレクトリに混在していてもよい。この場合、ファイル名文字列が画像データファイル以外のデータファイルであるか否かを判断し、画像データファイル以外のデータファイルであった場合は、ステップS101においてNOへ分岐すれば良い。なお、この場合に、データファイルが例えば文書データファイルであった場合は、その旨を示すアイコンを表示する様にしても良い。

【0091】[第4の実施形態] 第3の実施形態では、ディレクトリを表示する場合に、アイコンとディレクトリ名を表示することを説明したが、本実施形態では、アイコンとディレクトリ名とそのディレクトリの下に格納されているファイルの数を表示する場合を説明する。

【0092】本実施形態による動作を図14を用いて説明する。図14は第4の実施形態におけるディレクトリ表示処理の手順を表すフローチャートである。図中第3の実施形態と同様の処理を行う部分には、同一の番号を付けてある。

【0093】獲得したファイル名文字列がディレクトリ名を示す場合は、第3の実施形態で説明したようにステップS103からステップS104へ進み(図10)、図14のディレクトリ表示処理が実行される。第4の実施形態においてディレクトリを表示する場合、ディレク

10

20

30

40

50

トリ用のアイコン表示、ディレクトリ名の表示については、第1実施形態と同じである（ステップS120、S121）。第4の実施形態では、当該ディレクトリの下に格納されているファイルの数を数え、その数を表示するステップS122が追加されている。このステップS122では、ディレクトリを管理しているファイル（一般的なファイル管理システムはこのようなファイルを有する）をオープンし、そのディレクトリ管理のフォーマットに従って内容を読みだし、このディレクトリで管理されているJPEG符号化ファイル、JBIG符号化ファイル、MPEG符号化ファイルの数をカウントする。そして、このファイルの数をディレクトリ用アイコンと同時に表示する。

【0094】以上のように第4の実施形態によれば、各ディレクトリで管理しているファイルの数を把握することができ、ファイル検索の効率を上げることができる。

【0095】〔第5の実施形態〕上記第3及び第4の実施形態では、1つのディレクトリ内の画像データファイル、ディレクトリを表示している。第5の実施形態では、例えばキーワード検索等の検索処理の結果を表示するために、複数のディレクトリに跨って表示処理を行う場合を考える。この場合、例えば、キーワード検索の結果として得られる情報が図15に示すようなディレクトリ名のついたフルパスのファイル名である場合を説明する。これは、説明を簡単にするために、図9に示したディレクトリ構造のファイルの中から6個の画像ファイルを検索したものとする。

【0096】図16は第5の実施形態における検索結果表示処理の手順を示すフローチャートである。まずステップS130において、この処理で必要な初期化処理を行う。例えば、ディレクトリ名を比較するための、1つ前のディレクトリ名を格納する領域をクリアしたりする。次にステップS131で、検索結果の表示処理が終わったかを判断し、終わっていれば終了する。検索結果の表示処理が終了していない場合はステップS132に移る。

【0097】ステップS132では、検索結果のフルパスで格納されているファイル名からディレクトリ名を切り出す。次に、ステップS133において、この切り出したディレクトリ名が、現在表示しているディレクトリ名と同じか否かを判断し、同じであればステップS102に移り、検索された画像データファイルを表示する。同じでない場合には、ステップS134に移り、当該ディレクトリ名が既に表示されているか否かを判断する。

【0098】ステップS134において当該ディレクトリがまだ表示されていなければステップS104に移り、当該ディレクトリを表示する処理を行う。一方、当該ディレクトリが既に表示されていればそのままステップS131へ戻る。

【0099】例えば、図15に示した検索結果でこの処

理を行うと、“DIR-A/FILE1”については、ディレクトリ名“DIR-A”が切り出される（ステップS132）。この結果、今回表示するディレクトリは“DIR-A”となる。そして、ステップS102に進み、当該FILE1の画像データを表示する。

【0100】続く“DIR-A/FILE2”、“DIR-A/FILE4”については、今回のディレクトリ名と同じであるので、ステップS133からステップS102に進み、ファイル表示処理を行う。次に、“DIR-A/DIR1/FILE7”の場合、ステップS132で切り出されるディレクトリ名が“DIR-A/DIR1”になり、今回表示しているディレクトリ名（DIR-A）とは異なるためS134に移り、S104でディレクトリ表示処理を行う。続く、“DIR-A/DIR2/FILE9”も、同様にディレクトリ表示処理を行ってから、FILE9の画像表示を行う。“DIR-A/DIR2/FILE10”の場合は、ステップS134へ進むが、既にディレクトリ表示が行われているので、そのままステップS131へ戻る。

【0101】なお、上記第3～第5の実施形態では、白黒2値画像についてはJBIG符号化方式を、カラー画像についてはJPEG符号化方式を、動画像データについてはWindows-AVI符号化方式を使うことを説明したが、これらの符号化方式に限るものではなく、別の符号化方式でも良いことは容易に推察できる。

【0102】またJPEG符号化方式については、JPEG階層デコード方式により、検索表示画像を再生することを説明したが、通常のJPEGデコード方式でも良く、この場合には検索表示用の画像を原画像とは別に持つことも考えられる。更には、検索表示用の画像を持たない場合には、原画像を再生した後に縮小処理を行ってもよいことも、本実施形態の開示から容易に推察できる。

【0103】動画像符号化データの検索表示用の画像データは、その動画像データの先頭フレームを使うことを説明したが、これに限るものではなく、別のフレームでも良いことは、本実施形態の開示内容から容易に推察できる。

【0104】また、ディレクトリの場合、ディレクトリ用のアイコンを表示することを説明したが、これに限るものではない。検索表示の画面レイアウトもこれに限るものではない。

【0105】以上説明したように、第3～第5の実施形態によれば、ディレクトリ構造を有する画像ファイルを検索表示することにより、ディレクトリ構造をそのまま表現することができ、検索作業の効率を向上することができる。

【0106】また、ディレクトリ毎に検索画像を表示するため、1度に表示する検索画像の数が減り、検索効率を向上することができる。

10

20

30

40

50

【0107】また、ディレクトリ表示の際に、そのディレクトリで管理されている画像ファイルの数を表示することにより、検索効率を短縮することが可能になる。

【0108】〔第6の実施形態〕次に第6の実施形態を説明する。なお、第6の実施形態における画像データ管理装置の構成も第1の実施形態（図1）と同様であり、ここではその説明を省略する。

【0109】図17は第6の実施形態におけるアプリケーション実行の概念を説明する図である。図中、201は、本アプリケーションによって表示されているウィンドウを示している。本アプリケーションでは、符号化された画像データ・ファイルを読み込み、これをデコードして、ウィンドウ201内に表示する機能を有する。なお、本第6の実施形態では、説明を簡単にするため、画像データの符号化方式は、J P E G方式を用いる。202、203、204は画像表示枠であり、符号化された画像データ・ファイルをデコードし、各画像を表示しているところである。

【0110】206は他アプリケーションであり、ウィンドウ201を表示する本アプリケーションとは別のアプリケーションである。他アプリケーション206は、本アプリケーションと同様にJ P E G符号化データをデコード／表示する機能を有する。207は、他アプリケーション206が読み込む画像データ・ファイルである。また、205は他アプリケーション206がJ P E G符号化データ（画像データ・ファイル207）をデコードした結果の画像データである。208は、他アプリケーション206が出力した画像データファイル205の復号化画像データを本アプリケーションで表示するためにデータ変換処理を示している。

【0111】次に、第6の実施形態の動作について図を用いて説明する。まず本アプリケーションは、通常のJ P E G符号化データ・ファイルを読み込み、画像サイズがX、Yサイズ共に、1／8、1／4、1／2、1／1の種類のサイズの画像データを表示する機能を有する。そして、ユーザからファイル表示が指示されると、まずX、Yとも1／8の画像データをウィンドウに表示するように動作する。次に解像度アップの要求がくると、X、Yサイズ1／4の画像を、更に解像度アップの要求が来るとX、Yサイズ1／2の画像を、最終的には、1／1すなわち原画像を表示するように動作する。

【0112】図18は第6の実施形態における階層デコード／表示を実現するための機能構成を示すブロック図である。図18において、220はJ P E G符号化データ・ファイルである。22アはハフマン・デコード部であり、J P E G符号化データ・ファイル220をリードしてこれをデコードし、その結果、8×8のD C T係数222を得る。この後、通常は、8×8 R D C T 27を通して原画像を再生するのであるが、本アプリケーションでは、逆D C T部23に4種類のR D C T（逆D C

T）を備える。24は1×1 R D C T、25は2×2 R D C T、26は4×4 R D C T、27は8×8 R D C Tであり、これら各R D C T処理部を切り替えることにより、同一の符号化データからX、Yサイズ共に、1／8、1／4、1／2、1／1のサイズの画像228を再生することを可能にしている。

【0113】次に、J P E G符号化方式に複数種類の符号化方法が存在することを説明する。図19は、J P E G符号化方式における符号化アルゴリズムの種類を説明する図である。同図に示される様に、J P E G符号化方式で符号化された画像データ・ファイルには3種類のプロセスが有り、必須項目は253の「ベースライン・プロセス」のみである。また、この「ベースライン・プロセス」の中でも、各ヘッダ、セグメントの中で、符号化時にパラメータとして設定できる値がある。例えば、フレーム・ヘッダ内のコンポーネント数は、J P E Gアルゴリズム上は、1～255までの値を設定することが可能である。それらすべてのパターンをサポートするのは、現実的ではない。そのためデコードできないJ P E G符号化データ・ファイルが生ずることになる。

【0114】図20は第6の実施形態において保持されている画像データファイルの一例を示す図である。本実施形態の場合、図20の3種類のJ P E G符号化データ・ファイルの内、“F I L E 1. J P G”、“F I L E 2. J P G”は、本アプリケーションでデコード可能な符号化データ・ファイルであるとする。一方、“F I L E 3. J P G”は、本アプリケーションではデコードできないファイルであるとする。また、図17のウィンドウ201内に表示されている画像データ202、203、204は、“F I L E 1. J P G”、“F I L E 2. J P G”、“F I L E 3. J P G”に基づく3種類の画像を表示するものである。なお、各画像のX、Yサイズとも1／8の画像データであるとする。

【0115】従って、図17の画像データ202、203の画像を表示する処理は本アプリケーションで行い、画像データ204の画像を表示する処理は他アプリケーション206で行うことになる。この場合の動作を図21のフローチャートを用いて説明する。

【0116】図21は第6の実施形態における符号化画像データのデコード手順を示すフローチャートである。図中ステップS200では、指定された画像データ・ファイルをオープンする。ステップS201で、そのファイルの内容をリードし、各種ヘッダ、セグメントの内容を解析する。

【0117】図22はJ P E G符号化方式におけるデータ・シンタックスのフローを示す図である。図19で示した3種類のプロセスは、フレーム・ヘッダの識別子であるマーカー・コードを解析すれば良い。次に各テーブル、セグメント内に定義されている変数は、各テーブル、セグメントをそのフォーマットで読み、それらの値

10

20

30

40

50

を格納しておく。

【0118】ステップS202では、ステップS201で解析した結果にもとづき、本アプリケーションで当該JPEG符号化データをデコードできるかを判断する。このステップS202の判断は、フレーム・ヘッダのマーカー・コード、或いは各テーブル、セグメント内の変数の値に基づいて行われる。本アプリケーションでデコードが可能と判断されれば、ステップS204に移りデコードを行う。

【0119】一方、ステップS202の判断で、本アプリケーションではデコードできないと判断された場合には、ステップS203に移る。ステップS203では、当該JPEG符号化データ・ファイルをデコードするための予め用意された所定の別アプリケーションを起動して当該画像データファイルをデコードさせる。ステップS205では、別アプリケーションでデコードした結果の復号化画像データを本アプリケーションで取得する。ここで、この別アプリケーションでは、本アプリケーションが持っている階層デコード／表示機能は無く、原画像を再生する機能のみを有しているものとする。従って、他アプリケーションから取得される画像は、1/1の原画像となる。

【0120】ステップS206では、本アプリケーションが今どの階層の表示を行っているかを判断し、すなわちX、Yサイズが1/8、1/4、1/2、1/1のいずれであるかを判断し、その画像サイズに合うように、ステップS205で得た画像データを縮小するデータ変換処理である。ここで、画像データの縮小の方法としては、間引き処理等が挙げられる。ステップS207では、ステップS204もしくはステップS206で得られた画像データをウィンドウ201内に表示する。

【0121】以上のような第6の実施形態の動作例について、図20の画像データファイル構成を用いて更に詳しく説明する。図20で示す“FILE1. JPG”、“FILE2. JPG”は、本アプリケーションでデコード可能なJPEG符号化データ・ファイルであるため、この2つのファイルについてはステップS202の判断で、ステップS204に分岐する。ステップS204では、図18で示したように、1×1RDCT24を使い、X、Yとも1/8のサイズの画像データをデコードする。その後ステップS207に移り、デコードして得られた画像データを表示する。

【0122】次に“FILE3. JPG”を表示する要求がユーザから発行されると、ステップS202の判断で“NO”と判断され、ステップS203に分岐する。ステップS203では、例えば“PhotoShop”、“JPEGView”等、JPEG符号化データのデコード／表示機能を有するアプリケーションに“FILE3. JPG”を引渡し、当該符号化画像データのデコード処理を行わせる。この際、他アプリケーション

のファイル名、パス名等の設定は、予め本アプリケーションに設定されているものとする。また他アプリケーションを起動する場合には、バックグラウンドで起動し、他アプリケーションのウィンドウ等は、表示されないようにする。

【0123】このようにして、他アプリケーションにJPEG符号化データのデコード処理を行わせ、画像データを例えばメモリ（図1のRAM3）上に作らせる。そして、ステップS205では、このメモリ上に有る画像データを本アプリケーションが取得し、ステップS206へ移る。ステップS206では、現在、X、Yサイズ1/8の画像データを表示しているため、ステップS205で取得した画像データをX、Yサイズとも1/8に縮小する処理を行う。その後ステップS207に移り“FILE1. JPG”、“FILE2. JPG”と同様に“FILE3. JPG”を本アプリケーションのウィンドウ201内にあたかも本アプリケーションがデコード／表示したかのように表示することが可能となる。

【0124】なお、指定されたファイルがMPEG符号化方式で符号化された動画データの場合は次のようになる。ここで、本アプリケーションでは、MPEGデータはデコードできないものとする。その場合、図21のステップS202で、自分ではデコードできないと判断され、ステップS203に移る。その後、他アプリケーションで、指定されたMPEG符号化データをデコードし、画フレーム毎に本アプリケーションがデコードされたデータを取得し、本アプリケーションで表示を行う。

【0125】また、動画の表示の場合は、通常は、符号化データの最初からシーケンシャルに表示を行う。そのため、表示方法は本アプリケーションで行う場合と、他アプリケーションで行う場合で差が無い場合が多い。差が無い場合には、本アプリケーションが表示部分を制御する必要はなく、他アプリケーションでデコード／表示を行ってもよいことは明かである。

【0126】〔第7の実施形態〕第6の実施形態では、別のアプリケーションを予め指定しておき、その指定された別アプリケーションを実行することを説明したが、本第7の実施形態では、複数の別アプリケーションを実行する場合を説明する。第7の実施形態では、各別アプリケーションが符号化できたかどうかを判断するステップを設け、このステップにより、符号化データをデコードできるか、もしくは、実行している別アプリケーションがなくなるまで処理を繰り返す。

【0127】図23は、第7の実施形態における他アプリケーションによる画像データのデコード手順を示すフローチャートである。図23のフローチャートは、図21のステップS203を詳細に示すものである。ステップS210では、複数の他アプリケーションを実行するための初期化を行う。続いて、ステップS211では、他アプリケーションを実行する。ステップS212で

は、ステップS211で実行した他アプリケーションが当該画像データをデコードできたかを判断する。ここで、デコードができていれば、得られた画像データを本アプリケーションに引き渡すべくメモリ等に保持し、本処理を終了する。

【0128】一方、ステップS212において、当該画像データをデコードできなかった場合は、ステップS213へ進み、実行可能な他アプリケーションが他にあるかどうかを判定する。ステップS213で実行可能な更に他のアプリケーションがあればステップS211へ戻り、更に他のアプリケーションを用いて上述の処理を繰り返す。また、ステップS213で、実行可能な更に他のアプリケーションが存在しなければ、本処理を終了する。

【0129】以下、動作例を挙げて説明する。本例では、“App11”、“App12”、“App13”の3種類のアプリケーションがあり、図20の各画像データファイルをデコードするものとする。ここで、“App12”のみが、FILE3.JPGをデコードできるものとする。以下では、第6の実施形態と同じ動作をする部分についての説明は割愛し、他アプリケーションの実行制御部分のみ説明する。即ち、FILE3.JPGをデコードすべく指定された場合の、他アプリケーションによるデコード処理を説明する。

【0130】ステップS210で、例えば、“App11”、“App12”、“App13”の3種類のアプリケーションの実行ファイルのディレクトリ等を管理する。また実行順も決定する。次に、ステップS211に移り、まず1つ目のアプリケーション“App11”を実行する。実行後、ステップS212でその符号化データをデコードできたかの実行結果を判断する。“App11”は、FILE3.JPGをデコードできないので、ステップS213に移り、他のアプリケーションがあるかを判断する。この場合はまだあるのでステップS211に戻る。次に“App12”を実行する。“App12”は、FILE3.JPGデコードできるので、ステップS212の判断でこの処理から分岐し、処理を終了する。

【0131】他アプリケーションの決定法ならびに実行順などは、これに限るものではなく、例えば、他アプリケーションの決定は、本アプリケーションの立ち上げ時に決定してもよい。

【0132】また、他アプリケーションの検索方法も上記実施形態で説明した方法に限るものではない。また、上記第7の実施形態では、複数のアプリケーションを順次実行して符号化画像データのデコードを試みているが、ここでアプリケーションの順次実行とは、アプリケーションを順次起動するものでもよいし、複数のアプリケーションを予め起動しておいて、順次符号化画像データのデコードを行わせるものであってもよい。

【0133】また画像符号化方式は、JPEG符号化方式に限るものではなく、JBIG方式、その他静止画像符号化方式、あるいはMPEG等の動画像符号化方式でも良いことは容易に推察できる。

【0134】以上説明したように、実施形態6及び7によれば、自分のアプリケーションでサポートしている画像符号化方式で符号化されている画像符号化データ・ファイルについて他アプリケーションにデコードを行わせ、自身が提供する表示方式で表示することが可能となる。このため、自身の符号化画像データのデコード機能が十分でない場合でも、自アプリケーションの表示機能を用いて表示することが可能になる。

【0135】なお、本発明は、複数の機器（例えばホストコンピュータ、インタフェイス機器、リーダー、プリンタなど）から構成されるシステムに適用しても、一つの機器からなる装置（例えば、複写機、ファクシミリ装置など）に適用してもよい。

【0136】また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。

【0137】この場合、記憶媒体から読出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【0138】プログラムコードを供給するための記憶媒体としては、例えば、フロッピーディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモ리카ード、ROMなどを用いることができる。

【0139】また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0140】さらに、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0141】本発明を上記記憶媒体に適用する場合、その記憶媒体には、先に説明したフローチャートに対応す

るプログラムコードを格納することになるが、データファイルの記憶媒体への格納を管理するための制御プログラムについて簡単に説明すると、図 24 のメモリマップ例に示す各モジュールを記憶媒体に格納することになる。すなわち、少なくとも「発生処理モジュール」「付加処理モジュール」および「登録処理モジュール」の各モジュールのプログラムコードを記憶媒体に格納すればよい。

【0142】ここで、発生処理モジュールは、前記記憶媒体に格納されたデータファイルを特定する文字列を発生する発生処理を実現する。また、付加処理モジュールは、前記発生処理で発生した文字列に所望の文字列を付加する付加処理を実現する。更に、登録処理モジュールは、前記付加処理で得られた文字列を前記データファイルのファイル名として登録する登録処理を実現する。

【0143】更に、ディレクトリ構造で記憶媒体に格納された画像データファイルを管理するための制御プログラムについて簡単に説明すると、図 25 のメモリマップ例に示す各モジュールを記憶媒体に格納することになる。すなわち、少なくとも「判断処理モジュール」「第 1 表示処理モジュール」および「第 2 表示処理モジュール」の各モジュールのプログラムコードを記憶媒体に格納すればよい。

【0144】ここで、判断処理モジュールは、指定されたディレクトリに存在する名称データがファイル名かディレクトリ名かを判断する判断処理を実現する。また、第 1 表示処理モジュールは、前記判断処理により名称データがファイル名と判断された場合、当該ファイル名に対応する画像データファイルに基づいて画像を表示する第 1 表示処理を実現する。更に、第 2 表示処理モジュールは、前記判断処理により名称データがディレクトリ名と判断された場合、当該ディレクトリ名に対応する情報を表示する第 2 表示処理を実現する。

【0145】また、ディレクトリ構造で記憶媒体に格納された画像データファイルを管理するための制御プログラムの他の例について簡単に説明すると、図 26 のメモリマップ例に示す各モジュールを記憶媒体に格納することになる。すなわち、少なくとも「獲得処理モジュール」「判断処理モジュール」「第 1 表示処理モジュール」および「第 2 表示処理モジュール」の各モジュールのプログラムコードを記憶媒体に格納すればよい。

【0146】ここで、獲得処理モジュールは、所定の検索条件に適合する画像データファイルについて前記ディレクトリ構造に基づくディレクトリパス名の付加されたファイル名を獲得する獲得処理を実現する。また、判断処理モジュールは、前記獲得処理で獲得したファイル名に付加されたディレクトリパス名に基づいて、当該画像データファイルが現在表示中のディレクトリに存在するか否かを判断する判断処理を実現する。また、第 1 表示処理モジュールは、前記判断処理により前記画像データ

ファイルが現在表示中のディレクトリに存在すると判断された場合、該画像データファイルに基づいて画像を表示する第 1 表示処理を実現する。更に、第 2 表示処理モジュールは、前記判断処理により前記画像データファイルが現在表示中のディレクトリに存在しないと判定された場合、該画像データファイルが存在するディレクトリを示すディレクトリ情報を表示する第 2 表示処理を実現する。

【0147】また、符号化画像データを復号化して表示するための制御プログラムについて簡単に説明すると、図 27 のメモリマップ例に示す各モジュールを記憶媒体に格納することになる。すなわち、少なくとも「復号処理モジュール」「判断処理モジュール」「取得処理モジュール」および「表示処理モジュール」の各モジュールのプログラムコードを記憶媒体に格納すればよい。

【0148】ここで、復号処理モジュールは、符号化画像データを復号する復号処理を実現する。また、判断処理モジュールは、指定された符号化画像データを前記復号処理で復号化できるか否かを判断する判断処理を実現する。また、取得処理モジュールは、前記判断処理で復号化できないと判断された場合、所定のアプリケーションを起動し、該所定のアプリケーションに前記指定された符号化画像データを復号化させ、その結果の画像データを取得する取得処理を実現する。更に、表示処理モジュールは、前記取得処理で取得した画像データを表示する表示処理のコードとを備える。

【0149】

【発明の効果】以上説明した様に、本発明によれば、データファイルを蓄積する際に、データファイルの内容を示すキーワード等の情報を付加したファイル名を容易に登録できる。

【0150】また、本発明によれば、ファイル名として付与されたキーワードを、データファイルの検索情報として用いることが可能となり、ファイル検索がより容易に、かつ迅速に行える。

【0151】また、本発明によれば、複数の画像データファイルについて、各画像データファイル内の画像データに基づく表示を行うとともに、各画像データファイルが存在するディレクトリ名等のディレクトリ情報を表示することが可能となり、画像データファイルのディレクトリ構造を忠実に表示すると共に、各画像データを直接表示することが可能となる。

【0152】また、本発明によれば、アプリケーションがサポートしている画像符号化方式でデコードできない画像符号化データ・ファイルをデコード／表示する際に、その画像符号化データ・ファイルを別アプリケーションでデコードさせ、その結果を当該アプリケーションによって表示することが可能となり、当該アプリケーションが備えるユーザインターフェース機能等を有効に活用できる。

【0153】

【図面の簡単な説明】

【図1】第1の実施形態による画像データ管理装置の構成を示すブロック図である。

【図2】第1の実施形態における画像データファイルのファイル名のフォーマットを示す図である。

【図3】画像データを登録する際の操作画面例を示す図である。

【図4】第1の実施形態による画像データの登録手順を示すフローチャートである。

【図5】ファイル名の第0カラムへ登録すべき文字列を生成する手順を示すフローチャートである。

【図6】第2の実施形態における画像データファイルのファイル名のフォーマットを示す図である。

【図7】第2の実施形態における画像データ登録処理の手順を表すフローチャートである。

【図8】第3の実施形態における検索画像表示例を示す図である。

【図9】第3の実施形態におけるディレクトリ構造例を示す図である。

【図10】第3の実施形態における検索結果表示処理の手順を示すフローチャートである。

【図11】第3の実施形態におけるファイル表示処理の手順を表すフローチャートである。

【図12】第3の実施形態におけるディレクトリ表示処理の手順を表す図である。

【図13】J P E G 1 / 8 サイズのデコード処理の処理手順を説明する図である。

【図14】第4の実施形態におけるディレクトリ表示処

理の手順を表すフローチャートである。

【図15】第5の実施形態における画像データファイルの検索結果例を示す図である。

【図16】第5の実施形態における検索結果表示処理の手順を示すフローチャートである。

【図17】第6の実施形態におけるアプリケーション実行の概念を説明する図である。

【図18】第6の実施形態における階層デコード／表示を実現するための機能構成を示すブロック図である。

10 【図19】J P E G符号化方式における符号化アルゴリズムの種類を説明する図である。

【図20】第6の実施形態において保持されている画像データファイルの一例を示す図である。

【図21】第6の実施形態における符号化画像データのデコード手順を示すフローチャートである。

【図22】J P E G符号化方式におけるデータ・シンタックスのフローを示す図である。

20 【図23】第7の実施形態における他アプリケーションによる画像データのデコード手順を示すフローチャートである。

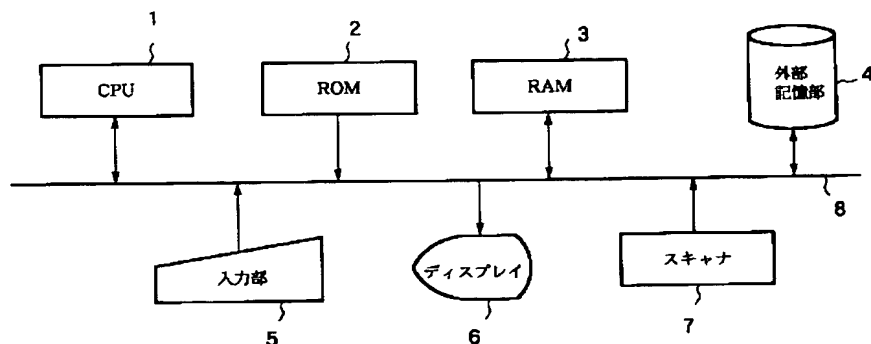
【図24】本発明に係る制御プログラムを格納する記憶媒体のメモリマップ例を示す図である。

【図25】本発明に係る制御プログラムを格納する記憶媒体のメモリマップ例を示す図である。

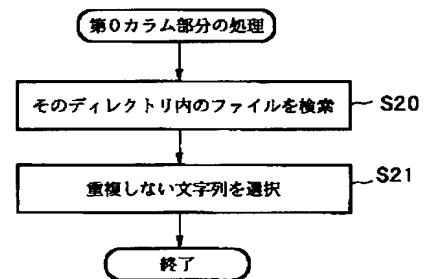
【図26】本発明に係る制御プログラムを格納する記憶媒体のメモリマップ例を示す図である。

【図27】本発明に係る制御プログラムを格納する記憶媒体のメモリマップ例を示す図である。

【図1】



【図5】



【図2】

	10 第0カラム	11 第1カラム	12 第2カラム	13 第3カラム	14 実際のファイル名
画像A	aaa0001	写真	子供	入学式	aaa0001_写真_子供_入学式.jpg
画像B	aaa0002	写真	子供	運動会	aaa0002_写真_子供_運動会.jpg
画像C	aaa0003	写真	子供	遠足	aaa0003_写真_子供_遠足.jpg
画像D	aaa0004	カタログ	旅行	ハワイ	aaa0004_旅行_ハワイ.jpg

【図3】

20 画像データ登録

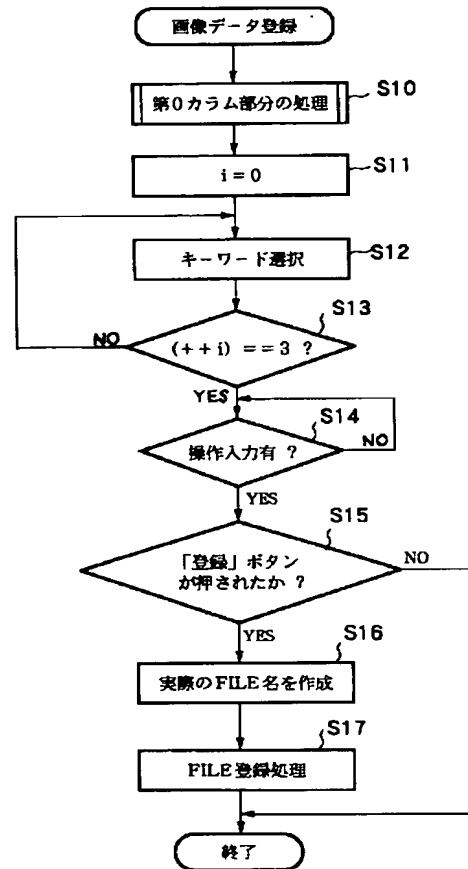
キーワード

写真	子供	入学式
カタログ	風景	遠足
文書	業務	運動会
録画

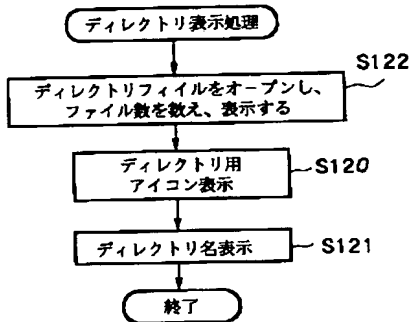
21

22 登録

【図4】



【図14】



【図15】

DIR - A / FILE1

DIR - A / FILE2

DIR - A / FILE4

DIR - A / DIR1 / FILE7

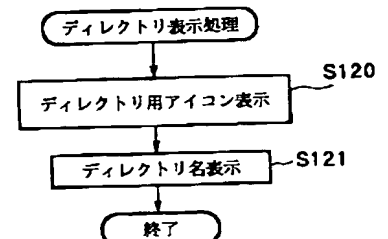
DIR - A / DIR2 / FILE9

DIR - A / DIR2 / FILE10

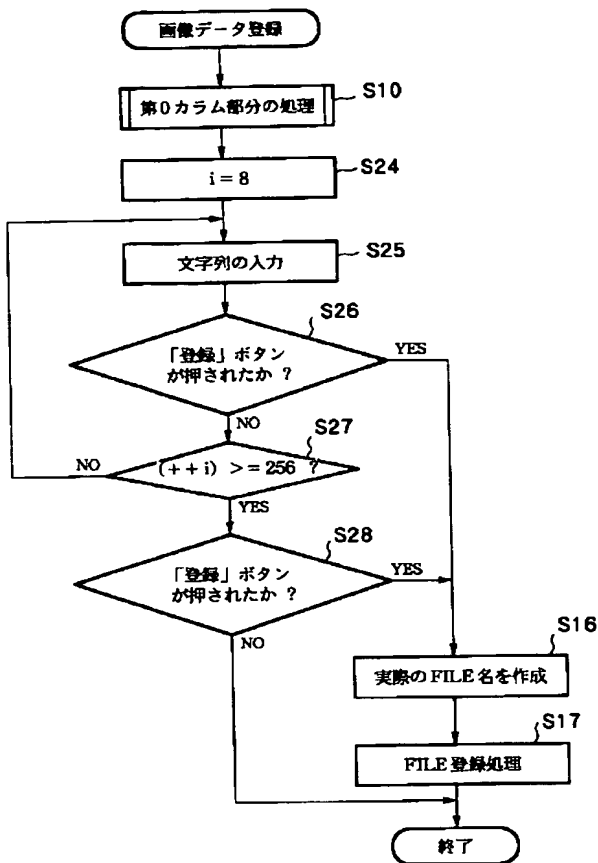
【図6】

	10 第0カラム	11 第1カラム	14 実際のファイル名
画像E	bbb0001	入学式の子供の写真	bbb0001_入学式の子供の写真
画像F	bbb0002	運動会の子供の写真	bbb0002_運動会の子供の写真
画像G	bbb0003	ハワイの旅行のカタログ	bbb0003_ハワイの旅行のカタログ

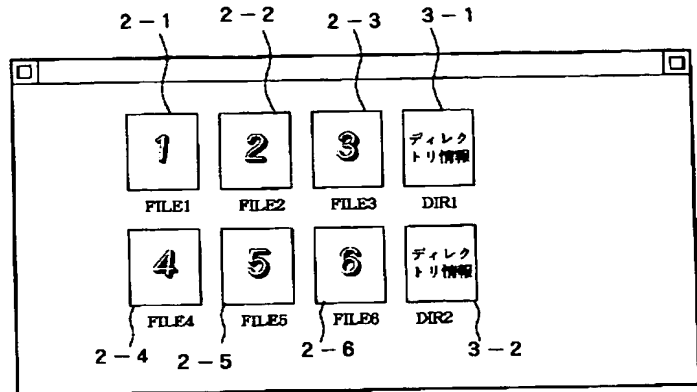
【図12】



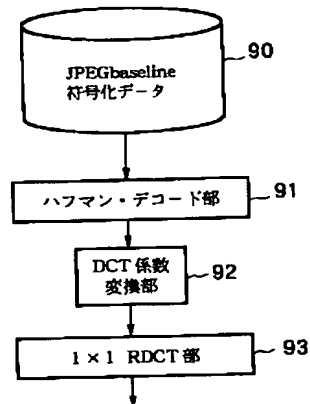
【図7】



【図8】

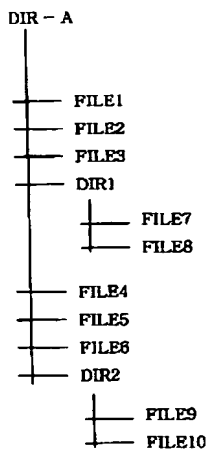


【図13】

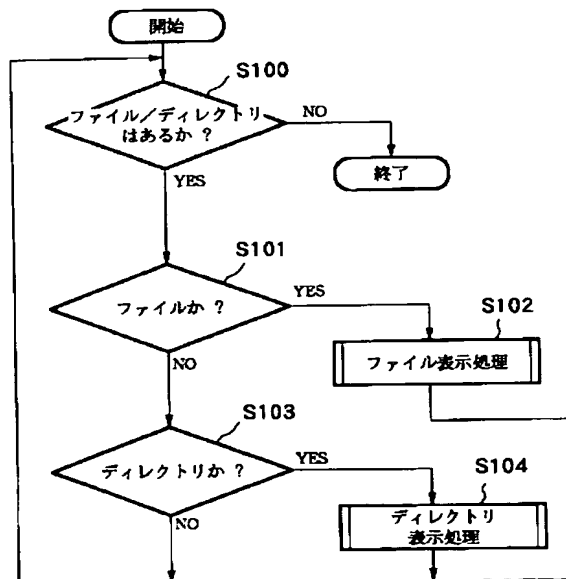


【図9】

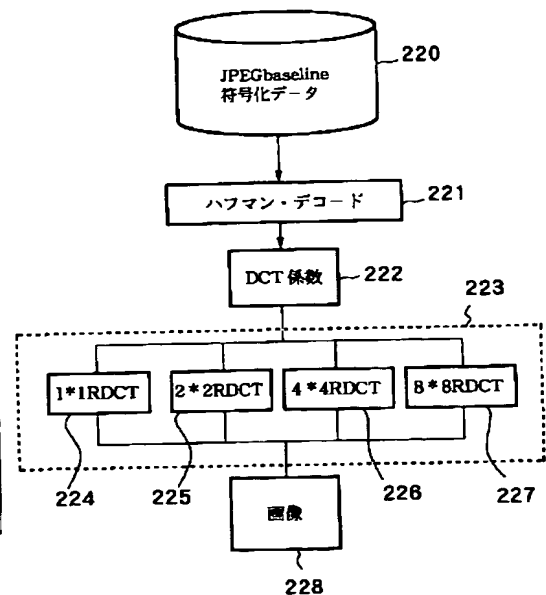
DIRECTORY構造



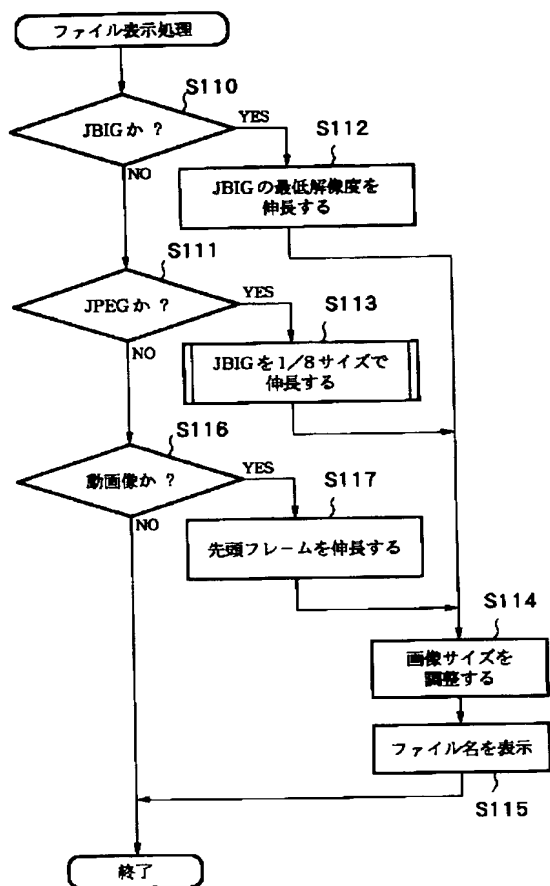
【図10】



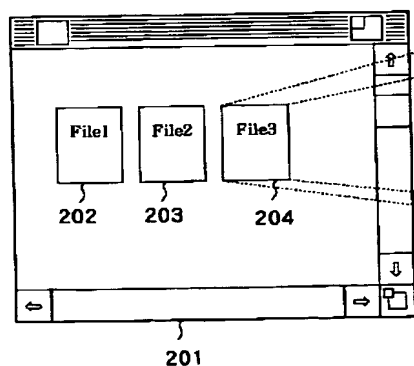
【図18】



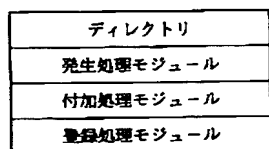
【図11】



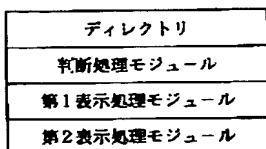
【図17】



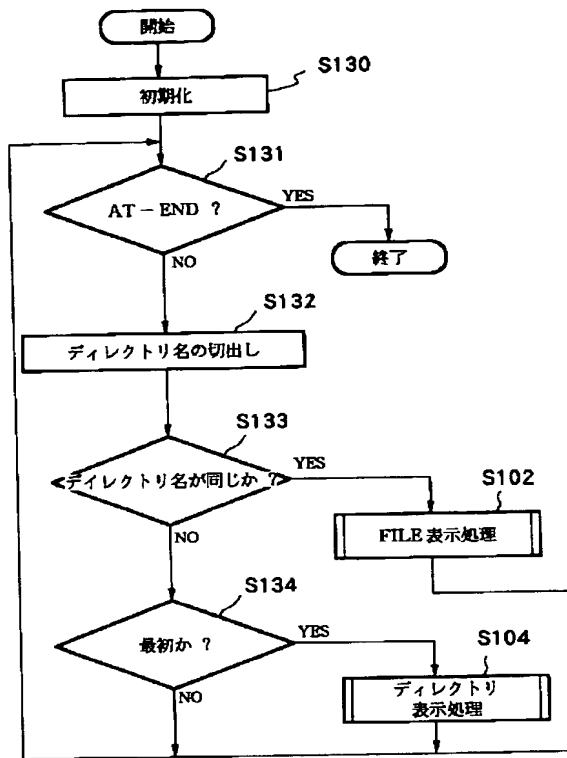
【図24】



【図25】



【図16】

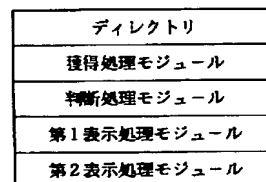


【図20】

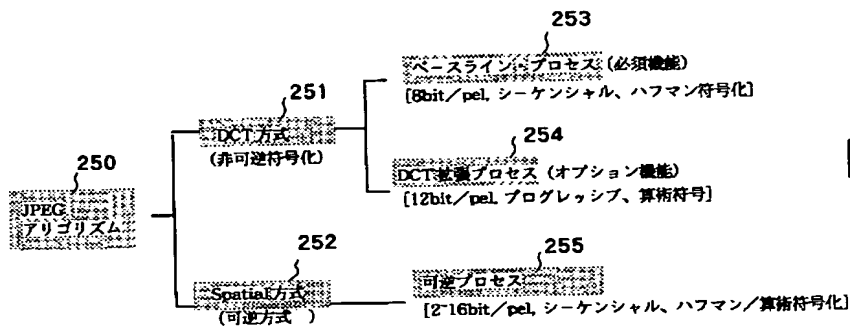
DIRECTORY - A

- FILE1.JPG (本アプリでDecode可能)
- FILE2.JPG (本アプリでDecode可能)
- FILE3.JPG (本アプリではDecode不可能)

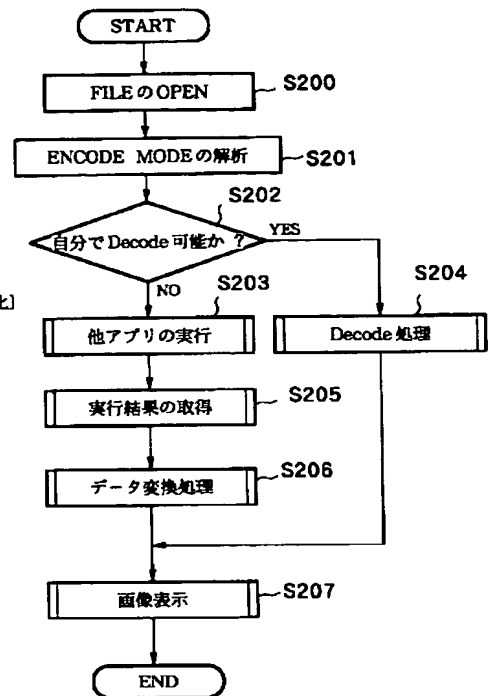
【図26】



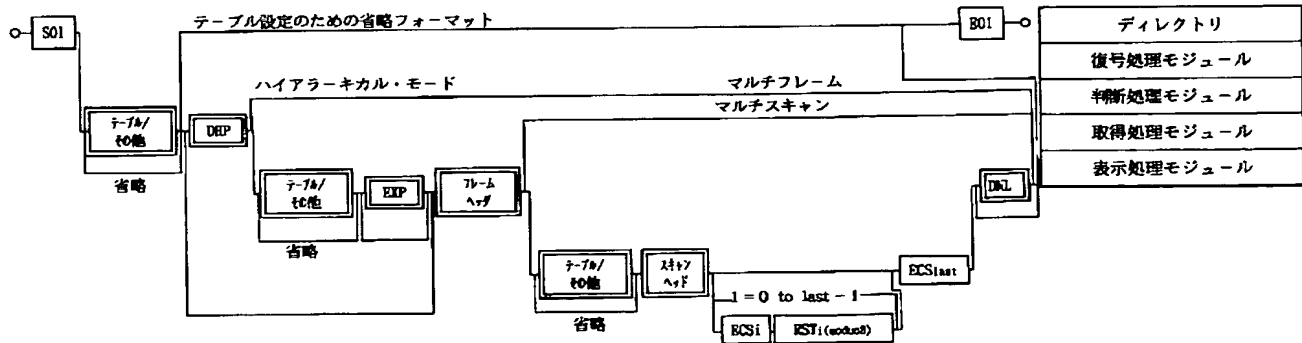
【図19】



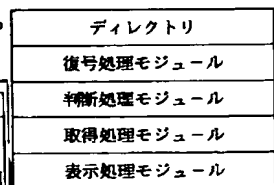
【図21】



【図22】



【図27】



【図 2 3】

